

Vertex Importance Extension of Betweenness Centrality Algorithm

Jiří Hanzelka^(1,2), Michal Běloch⁽¹⁾, Jan Martinovič⁽¹⁾, Kateřina Slaninová⁽¹⁾

¹ IT4Innovations, VŠB – Technical University of Ostrava,
17. listopadu 15/2172, 708 33 Ostrava – Poruba, Czech Republic
{jiri.hanzelka, michal.beloch, jan.martinovic, katerina.slaninova}@vsb.cz

² Department of Computer Science, FEECS, VŠB – Technical University of Ostrava,
17. listopadu 15/2172, 708 33 Ostrava – Poruba, Czech Republic
jiri.hanzelka@vsb.cz

Abstract. Variety of real-life structures can be simplified by a graph. Such simplification emphasizes the structure represented by vertices connected via edges. A common method for the analysis of the vertices importance in a network is betweenness centrality. The centrality is computed using the information about the shortest paths that exist in a graph. This approach puts the importance on the edges that connect the vertices. However, not all vertices are equal. Some of them might be more important than others or have more significant influence on the behavior of the network. Therefore, we introduce the modification of the betweenness centrality algorithm that takes into account the vertex importance. This approach allows the further refinement of the betweenness centrality score to fulfill the needs of the network better. We show this idea on an example of the real traffic network. We test the performance of the algorithm on the traffic network data from the city of Bratislava, Slovakia to prove that the inclusion of the modification does not hinder the original algorithm much. We also provide a visualization of the traffic network of the city of Ostrava, the Czech Republic to show the effect of the vertex importance adjustment. The algorithm was parallelized by MPI¹ and was tested on the supercomputer Salomon² at IT4Innovations National Supercomputing Center, the Czech Republic.

Keywords: betweenness centrality, high performance computing, MPI, traffic network

1 Introduction

The graph theory is proving to be a useful tool in the variety of research fields. A graph can describe objects and their communication and/or relationship. This has become very advantageous in areas like biology and chemistry [1,2,3], sociology [4,5,6,7], linguistics [8,9], physics and informatics [10,11,12,13,14]. The structure and topology of the graph or network thus become the focus of the research. Although the meaning of vertices and edges is different across the scientific fields, the underlying idea remains the same. For example, the interaction of proteins in biology is not different to an interaction of two people in a social network. Proteins and people are represented by vertices and their interaction by edges. It is the weight of an edge that allows the quantification of the interaction.

The shared interest is finding the important or influential vertex in a given graph. For this purpose, the various centrality scores are capable of an answer. There is a plethora of them

¹ <http://www.mpi-forum.org/>

² <https://docs.it4i.cz/>

available, like closeness, degree, eigenvector or Katz centrality, but the most common and widely adopted is the betweenness centrality score [15]. The role of betweenness centrality score is to identify the busiest vertices in the network. This score is computed using the knowledge of the shortest paths in the network. Therefore, it adopts the information from the edges. But the vertices also contain valuable information that should not be neglected.

Particularly, interesting and growing area where the graph theory has its place is the urban traffic monitoring. The latest development of the society sees the increase of the vehicles in the cities as more people are able to afford a car. Therefore, the advanced planning of the infrastructure and the traffic control become more important. One approach is to make a connection with the known physical problem. Daganzo [16] is treating traffic flow as a hydrodynamic model and solves it using differential equations. More general approach to traffic flow is based on the application of extended Kalman filter [17, 18]. Other authors noted the trend pattern in traffic flow and analyzed it as time series [19, 20]. Since the traffic flow is happening on the roads, it is natural to consider the road network as well. The graph theory helps in visualization of the transportation network, where the vertices represent crossroads, turnings, exits from the parking lot etc., and edges represent roads. The oriented weighted graph makes the faithful description of the real network. Additionally, the weights of the edges can be used to represent various information about the conditions of a given section like speed, weather condition, time of the day, etc.

The graph of the city transportation network is complicated even for smaller cities. Therefore, it is desirable to extract as much information as possible from it. The common way is to use the betweenness centrality score that gives relevant answers about the vertex importance. The role of betweenness centrality score is to identify the busiest vertices in the network. This approach is examined by authors in [21, 22, 23, 24]. However, some authors argue that human rationality is bounded [25] and their choice of the shortest path is not equivalent to the topological choice. Another problem is a high dynamic of the network. Thus, the authors raised the question whether betweenness alone is enough to describe the traffic flow. While the concern is certainly valid, there is still a lot of research to be done on the topic.

For our purpose of traffic monitoring, the betweenness centrality is satisfactory. The problem we discovered is that the betweenness centrality uses the information about the shortest paths in the graph and it does not take into account the vertices which are passed when going from the origin to the destination. People do not use the network just for the sake of using it. They enter it at some origin point because they care for the destination. Let us consider a major event is happening in the city like a concert or political meeting. We can expect increased traffic towards this vertex when the people want to reach this event and when the people are leaving it. We could change the weights of all edges lying on the shortest paths to the destination to account the increase in traffic, but it would require the change of a significant amount of values. Moreover, when the people will leave the event, it would be difficult to assess which values to increase, since the destination points are unknown. Thus, we propose the assignment of the weight to the vertex. It can also help solve the dynamic problem mentioned above.

While the motivation behind the modification was its use in the traffic network, we believe its usefulness goes beyond it. While analyzing the social behavior of a group, the ability to increase the importance of a given vertex (individual) can be used to describe a sudden change in the group dynamics. When studying the spread of information in a network, credible and dependable sources can have increased importance. These are just few examples where the concept of the vertex importance can be considered.

This paper is organized as follows. Section 2 briefly reminds of the betweenness centrality. Section 3 describes the proposed modification. Section 4 shows the results of the performance and Section 5 shows the visualization and examples, both carried out on real traffic data. Section 6 concludes the paper.

2 Betweenness centrality

Betweenness centrality or in full form Shortest path betweenness centrality first appeared in sociology [26]. Its main purpose was to quantify the individual's influence over the information flow in the social network. It has been generalized and adopted since as one of the centrality measures in the network. The higher value of betweenness centrality means the greater importance within the network. Specifically, within the context of the traffic network it usually represents a problematic section of the traffic flow.

Betweenness centrality of a vertex is defined as the ratio of the number of the shortest paths between an origin and a destination that pass through the vertex and the number of all the shortest paths between an origin and a destination.

Let us describe the traffic network with the graph $G=(V,E)$, where V is the set of vertices pair-wise connected by edges forming the set E . We define a path from $s \in V$ to $t \in V$ as an alternating sequence of vertices and edges, beginning with s and ending with t , such that each edge connects its preceding vertex with its succeeding vertex. The symbol $\sigma_{st} = \sigma_{ts}$ denotes the number of the shortest paths between vertex $s \in V$ and $t \in V$, and $\sigma_{st}(v)$ is the number of the shortest paths between s and t that goes through $v \in V$. Betweenness centrality BC for the vertex $v \in V$ is then defined as:

$$BC(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (1)$$

To get the shortest paths we use standard Dijkstra algorithm [27]. Betweenness centrality is computed using Brandes algorithm [28]. The pseudo code is mentioned below as Algorithm 1.

```

input : directed graph  $G = (V, E)$  with edge lengths  $\lambda : E \rightarrow \mathbb{R} > 0$ 
data : priority queue  $Q$  with keys  $dist[\cdot]$ 
        stack  $S$  (both initially empty) and for all  $v \in V$ 
         $dist[v]$ : distance from source
         $Pred[v]$ : list of predecessors on shortest paths from source
         $\sigma[v]$ : number of shortest paths from source to  $v \in V$ 
         $\delta[v]$ : dependency of source on  $v \in V$ 
output: betweenness  $c_B[v]$  for all  $v \in V$  initialized to 0
1 foreach  $s \in V$  do
2   foreach  $w \in V$  do  $Pred[w] \leftarrow$  empty list ;
3   foreach  $t \in V$  do  $dist[t] \leftarrow \infty$ ;  $\sigma[t] \leftarrow 0$  ;
4    $dist[s] \leftarrow 0$ ;  $\sigma[s] \leftarrow 1$  enqueue  $s \rightarrow Q$ 
5   while  $Q$  not empty do
6     extract  $v \leftarrow Q$  with minimum  $dist[v]$ ; push  $v \rightarrow S$ 
7     foreach vertex  $w$  such that  $(v, w) \in E$  do
8       if  $dist[w] = dist[v] + \lambda(v, w)$  then
9          $dist[w] \leftarrow dist[v] + \lambda(v, w)$ 
10        insert/update  $w \rightarrow Q$  with new key
11         $\sigma[w] \leftarrow 0$ 
12         $Pred[w]$  gets empty list
13      end
14      if  $dist[w] = dist[v] + \lambda(v, w)$  then
15         $\sigma[w] \leftarrow \sigma[w] + \sigma[v]$ 
16        append  $v \rightarrow Pred[w]$ 
17      end
18    end
19  foreach  $v \in V$  do  $\delta[v] \leftarrow 0$  ;
20  while  $S$  not empty do
21    pop  $w \leftarrow S$ 
22    foreach  $v \in Pred[w]$  do  $\delta[v] \leftarrow \delta[v] + \frac{\sigma[v]}{\sigma[w]} \cdot (1 + \delta[w])$  ;
23    if  $w \neq s$  then  $c_B[w] \leftarrow c_B[w] + \delta[w]$  ;
24 end

```

Algorithm 1. Original betweenness centrality for directed graph with edge lengths

3 Modification

As mentioned in the introduction, the purpose of this modification is to describe the situation better, when the importance will change for one vertex or some number of vertices.

We identified that there are two natural types of events that can occur in experiments or real situations. The first event arises when the vertex is the source of the shortest paths. It means every

shortest path leading from this point is influenced by the increase. To simulate this event, we selected the parameter *alpha* to describe the rise of the vertex importance.

The second event arises when the vertex is more important as a destination point for the betweenness calculation. This kind of behavior influences all the shortest paths that lead to the destination vertex. For this phenomenon, we designated the parameter *beta*.

Therefore, the importance parameter of the vertex in a modified version of the betweenness centrality algorithm consists of two positive real numbers *alpha* and *beta*.

- *alpha* $\in \mathbb{R} \geq 0$ (source vertex importance) - increases betweenness on the shortest paths directing from this vertex
- *beta* $\in \mathbb{R} \geq 0$ (destination vertex importance) - increases betweenness on the shortest paths directing to this vertex

The betweenness gain on the shortest paths is proportional to the chosen values. For example, if *alpha* is equal to 2 for vertex *A* ($\alpha_A = 2$), then betweenness centrality gain on vertices that are on the shortest paths from vertex *A* to all other vertices is twice as big. This situation is illustrated in Figure 1. The betweenness centrality is calculated just from the vertex *A* to all other vertices and the red edge means that it is lying on the shortest path from vertex *A* to all other vertices.

Change in *beta* value can be seen in Figure 2, where betweenness centrality is also calculated just from vertex *A* to all other vertices. We set *beta* equal to 2 for vertex *D* ($\beta_D = 2$). In this case, the betweenness is doubled for every surrounding vertex. The amount of betweenness gain for vertices that are further from the destination vertex slowly dissipates.

The pseudo code of this modification can be seen in Algorithm 2. The modification is on line 23 where we added the parameters *alpha* and *beta*. If *alpha* and *beta* values are set to value 1, we get original unmodified algorithm from Ulrik Brandes as seen in Algorithm 1.

```

19  |  foreach  $v \in V$  do  $\delta[v] \leftarrow 0$  ;
20  |  while  $S$  not empty do
21  |  |   $\text{pop } w \leftarrow S$ 
22  |  |  foreach  $v \in \text{Pred}[w]$  do
23  |  |  |   $\delta[v] \leftarrow \delta[v] + \frac{\sigma[v]}{\sigma[w]} \cdot (\alpha[s] \cdot \beta[w] + \delta[w])$ 
24  |  |  end
25  |  |  if  $w \neq s$  then  $c_B[w] \leftarrow c_B[w] + \delta[w]$  ;

```

Algorithm 2. Modification of original betweenness centrality for directed graph with edge lengths

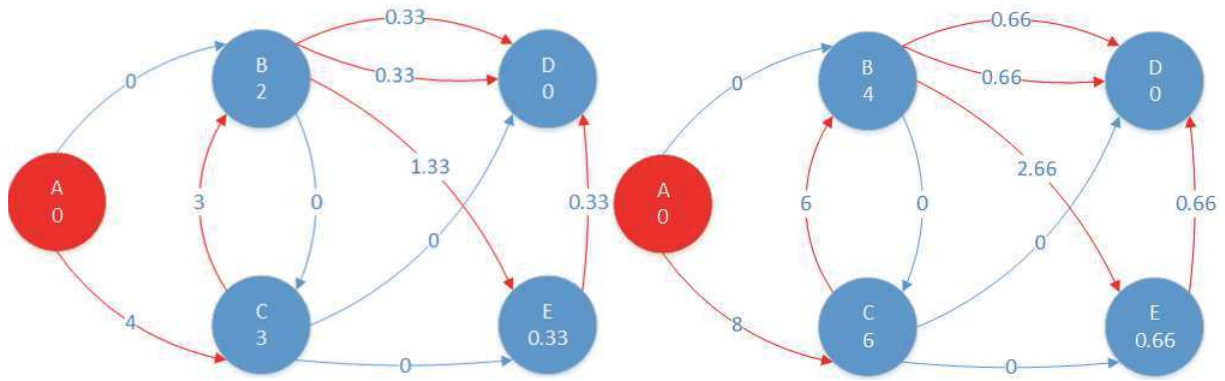


Fig. 1. Comparison of betweenness gain for vertex A, when $\alpha_A = 1$ on the left and $\alpha_A = 2$ on the right

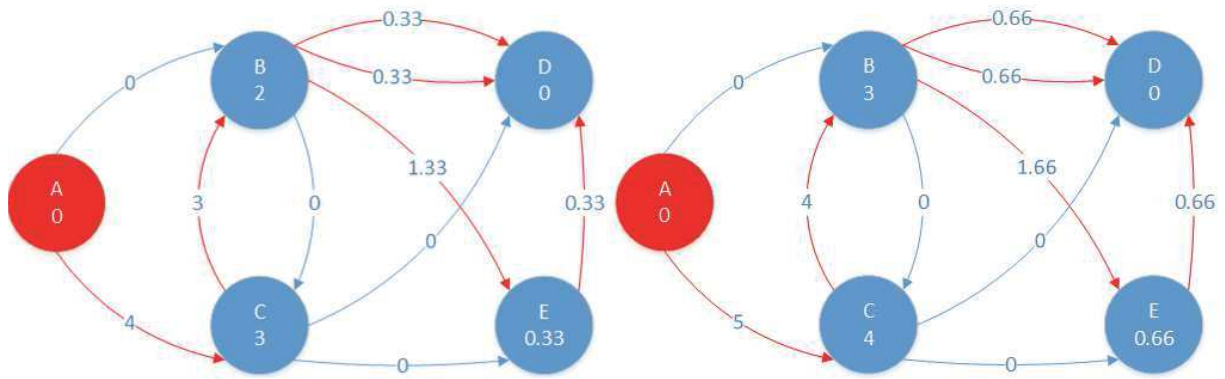


Fig. 2. Comparison of betweenness gain for vertex A, when $\beta_D = 1$ on the left and $\beta_D = 2$ on the right

4 Performance test on a traffic network

To test our modification, we chose a graph of the traffic network. This experiment was focused on measuring the difference in speeds and memory requirements of the original and modified betweenness centrality algorithm parallelized using MPI on a graph generated from the actual traffic network of Bratislava city in Slovak Republic as an input. The source graph contained 225,526 vertices and 529,767 edges. Data are obtained from open source OpenStreetMap project³.

The performance was tested on Salomon cluster operated by IT4Innovations National Supercomputing Center. The cluster consists of 1,008 compute nodes and each of them contains 2x Intel Xeon E5-2680v3 processors clocked at 2.5 GHz and 128 GB DDR4@2133 RAM. When the experiments were performed, the operating system was CentOS 6.7 and the code was compiled using Intel C++ Compiler 17.0. To increase the speed for a given architecture, *-xHost* flag was

³ <https://www.openstreetmap.org>

used in a compilation. This flag allows to use the highest instruction set available on our Haswell architecture.

Table 1. Comparison of the original and modified version of the betweenness centrality algorithm on the graph of Bratislava city traffic network

Computing Nodes	MPI processes	Speed (s)		Memory requirements per process (MB)		Speed decrease (%)	Memory increase (%)
		Orig. BW	Mod. BW	Orig. BW	Mod. BW		
1	24	812.21	883.35	86.0	89.5	8.76	4.07
2	48	414.74	452.91	95.4	98.9	9.20	3.67
4	96	208.45	226.52	88.5	92.0	8.67	3.95
8	192	104.88	113.68	89.0	93.0	8.39	4.49

Table 1 shows the speed and memory requirements of the original and modified version of the betweenness centrality algorithm. The test was performed for a different number of computing nodes from one to eight. On each computing node, a process per processor core was created. So, one node could have 24 processes, because each node had 2x12 cores. On average, each process consumed 89.73 MB of RAM for the original version of the algorithm and 93.35 MB for the modified version. The modified version of algorithm took on average about 8.76 percent longer when compared to the original version. Speed decrease (SD) was calculated according to formula $SD = (t_2 - t_1)/t_1 * 100$, where t_1 is time of original version of betweenness and t_2 is time of modified version of betweenness. We also tested the memory requirements of both versions of the algorithm. From the results, we can compute that the modified version of algorithm required around 4.05 percent more memory. To summarize, the overhead of modified version is acceptable for us.

5 Experiments

To better describe and provide visual example of our modification, we chose a graph of a traffic network again because it is easy to interpret and visualize. For this particular case, we can notice certain ambiguity between parameters *alpha* and *beta*. Consider the example of people going to work from certain suburb area in the morning. Do we increase the parameter *alpha* to show the increase in traffic coming from the source or parameter *beta* to show increase in traffic going to the destination? We usually want to increase both values, but we should base this on our knowledge of the local network and people's behavior. Therefore, this change should be used in the context of macro-simulation when *alpha* and *beta* values are selected in agreement with the analysis of mobility of the population. If we get the origin-destination matrix, we can use it to adjust these values for each vertex and the resulting betweenness can be more precise. We can also simulate differences of betweenness for different times on the same graph of the traffic network. Therefore,

we can identify bottlenecks during weekdays in the morning, when a lot of people are going to work, or in the evening when they are going back home.

Our modified version of the betweenness centrality was integrated into the Floreon+ system⁴ operated by IT4Innovations National Supercomputing Centre⁵. The objectives of the Floreon+ system are monitoring, modelling and prediction of crisis situations. With the help of this system, we created examples of betweenness centrality algorithm results when we modified *alpha* and *beta* values. We used different values to illustrate the contrast between them and the default values. The subset of a real traffic network of Ostrava city, the Czech Republic was used as an input graph. The dimension of the observed area was 45x60 km. The visualization of the betweenness centrality is therefore confined to this area.

You can see the selected area of traffic network of Ostrava city in Figure 3. Betweenness centrality score was obtained via the unmodified algorithm (we used default values $\alpha = 1$, $\beta = 1$ in this example). The width of the red line denotes the value of betweenness score; the wider the line is, the bigger the betweenness centrality values are. At the bottom of this picture, there is a black circle with the selected vertex for which *beta* value will be changed.

Figure 4 shows the value of *beta* increased to 5,000 for the selected vertex in the black circle. This change makes the vertex more important destination for the traffic. We can see a visible change of the line width directing to this vertex.

Figure 5 shows another example. The selected part of Ostrava city remains the same, but this time we consider three points of interest. Here we use the original betweenness centrality, i.e. $\alpha = 1$ and $\beta = 1$ for all the circles.

Then Figure 6 shows how the betweenness changes in comparison to Figure 5, when we set the value of *beta* to 2,000 for the black circles and the value of *alpha* to 5,000 for the red circle. This change is clearly visible and simulates higher traffic around the red circle where the selected vertex is more important as a source point for the shortest paths. Also, higher traffic is visible on main road between black circles, where the selected vertices are more important as a destination points.

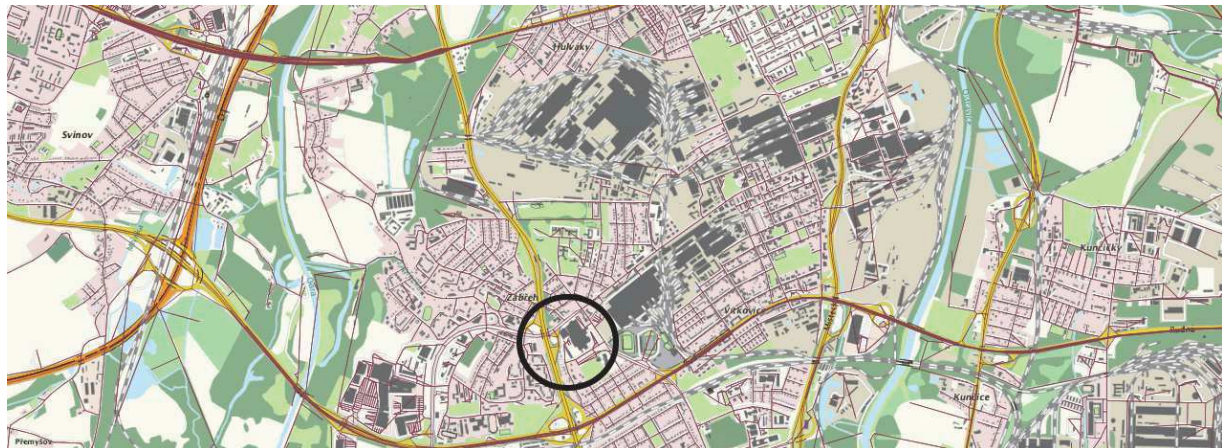


Fig. 3. Visualized betweenness centrality output for a part of Ostrava city. In black circle, there is a selected vertex with value $\beta = 1$.

⁴ <https://floreon.it4i.cz>

⁵ <http://www.it4i.eu>

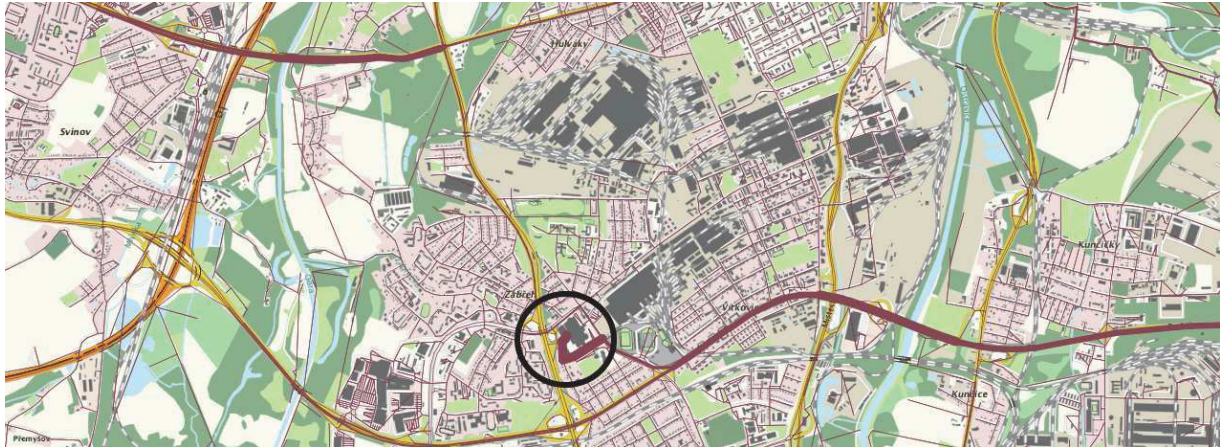


Fig. 4. Visualized betweenness centrality output for a part of Ostrava city. In black circle, there is a selected vertex with value $\beta = 5,000$.



Fig. 5. Visualized betweenness centrality output for a part of Ostrava city. In black circles, there are selected vertices with value $\beta = 1$, and in red circle there is a selected vertex with value $\alpha = 1$.



Fig. 6. Visualized betweenness centrality output for a part of Ostrava city. In black circles, there are selected vertices with value $\beta = 2,000$, and in red circle there is a selected vertex with value $\alpha = 5,000$.

6 Conclusion

In this paper, we described the modification of betweenness centrality algorithm from Ulrik Brandes, so it is able to take into calculation the vertex importance in the graph. We identified two types of vertex importance – source vertex (α) and destination vertex (β) importance.

The main objective of this change was to add new possibilities to work with betweenness centrality. We chose graphs of the traffic network for the testing the algorithm. This type of network is very static, but the events occurring here are on the other hand very dynamic. With this modification, we can easily adjust the betweenness score during the day cycle, and we can also quickly react to new events. It also allows us to anticipate problematic parts and bottlenecks in the traffic flow better.

The comparison of the original betweenness algorithm and our modified version shows favorable results for our modification. When they were both tested on the same traffic network depicting the city of Bratislava, the additional computation slowed the algorithm on average by 8.76 %. The memory consumption increase was only 4.05 %. The performance of the modification is therefore very good.

The attached visualization of the Ostrava city traffic network shows the interpretation of these two values and how they differ from each other.

The only thing we have not tackled in this paper is how to choose the values themselves. There is no quick and easy recommendation. It is imperative to know the structure of your network. It is important to perform the research on the mobility of the population for the particular traffic network. The origin – destination matrix can be used for more precise adjustment of the parameters. Monitoring of the network is also very useful to get information on movements within the network during the day cycle.

The future research will be focused on better understanding of the α and β values. We also will focus on better analysis of our traffic network so we can better refine these coefficients.

Developing self-sustaining system capable of auto adjustment depending on the current situation is one of the goals of the project ANTAREX⁶ and its use case focused on self-adapted server side/client side navigation system. The proposed extension of the betweenness centrality algorithm will be tested within this project on the server-side navigation system part to obtain a global view of traffic network and to be able to influence the traffic flow within the Smart City context.

Acknowledgment

This work was supported by The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project ‘IT4Innovations excellence in science - LQ1602’, by the IT4Innovations infrastructure which is supported from the Large Infrastructures for Research, Experimental Development and Innovations project ‘IT4Innovations National Supercomputing Center – LM2015070’, and partially by ANTAREX, a project supported by the EU H2020 FET-HPC program under grant 671623, and by grant of SGS No. SP2017/182 “Solving graph problems on spatio-temporal graphs with uncertainty using HPC”, VŠB - Technical University of Ostrava, Czech Republic.

References

1. LI, M., WANG, J., CHEN, X., WANG H., PAN, Y. A local average connectivity-based method for identifying essential proteins from the network level. *Computational Biology and Chemistry* **35** (2011) 143 – 150
2. XIA, J., SUN, J., JIA, P., ZHAO, Z. Do cancer proteins really interact strongly in the human protein–protein interaction network? *Computational Biology and Chemistry* **35** (2011) 121 – 125
3. HAGMANN, P., CAMMOUN, L., GIGANDET, X., MEULI, R., HONEY C.J., WEDEEN, V.J., SPORNS O., FRISTON, K.J. Mapping the Structural Core of Human Cerebral Cortex. *PLoS Biology* **6** (2008) 1479 – 1493
4. EVERETT, M.G., BORGATTI, S.P., The centrality of groups and classes. *Journal of Mathematical Sociology* **23** (1999) 181 – 201
5. SZELL, M., THURNER, S. Measuring social dynamics in a massive multiplayer online game. *Social Networks* **32** (2010) 313 – 329
6. WASSERMAN, S., FAUST, K. Social network analysis: Methods and applications. Cambridge University Press., Cambridge, England (1994)
7. CLIFTON, A., TURKHEIMER, E., OLTMANNS, T.F. Personality disorder in social networks: Network position as a marker of interpersonal dysfunction. *Social Networks* **31** (2009) 26 – 32
8. VANDENBERGHE, R., WANG, Y., NELISSEN, N., VANDENBULCKE, M., DHOLLANDER, T., SUNAERT, S., DUPONT, P. The associative-semantic network for words and pictures: Effective connectivity and graph analysis. *Brain and Language* **127** (2013) 264 – 272
9. HE, T., ZHAO, J., LI, J. Discovering relations among named entities by detecting community structure. *PACLIC20* (2006) 42 – 48

⁶ <http://www.antarex-project.eu/>

10. DONGES, J. F., ZOU, Y., MARWAN, N., KURTHS, J. The backbone of the climate network. *EPL (Europhysics Letters)* **87** (2009)
11. ZHANG, G.Q., WANG, D., LI, G.J. Enhancing the transmission efficiency by edge deletion in scale-free networks. *Physical Review E* **76** (2007)
12. ZHOU, S., MONDRAGÓN, R.J. Accurately modeling the internet topology. *Physical Review E* **70** (2004)
13. COMIN, C.H., DA FONTOURA COSTA, L. Identifying the starting point of a spreading process in complex networks. *Physical Review E* **84** (2011)
14. KAWAMOTO, H., IGARASHI, A. Efficient packet routing strategy in complex networks. *Physica A: Statistical Mechanics and its Applications* **391** (2012) 895 – 904
15. KLEIN, D.J., Centrality measure in graphs. *Journal of Mathematical Chemistry* **47** (2010) 1209 – 1223
16. DAGANZO, C.F. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological* **28** (1994) 269 – 287
17. WANG, Y., PAPAGEORGIOU, M. Real-time freeway traffic state estimation based on extended Kalman filter: a general approach. *Transportation Research Part B: Methodological* **39** (2005) 141 – 167
18. NGODUY, D. Real time multiclass traffic flow modelling-English M25 freeway case study. *12th Conference of the Hong-Kong Society for Transportation Studies* (2007) 143 – 152
19. SHANG, P., LI, X., KAMAE, S. Chaotic analysis of traffic time series. *Chaos, Solitons & Fractals* **25** (2005) 121 – 128
20. HONG, WC., DONG, Y., ZHENG, F., LAI, CY. Forecasting urban traffic flow by SVR with continuous ACO. *Applied Mathematical Modelling* **35** (2011) 1282 – 1291
21. GALAFASSI, C., BAZZAN, A.L.C. Analysis of Traffic Behavior in Regular Grid and Real World Networks (2013)
22. KAZERANI, A., WINTER, S. Can betweenness centrality explain traffic flow? *12th AGILE International Conference on Geographic Information Science* (2009) 1 – 9
23. GAO, S., WANG, Y., GAO, Y., LIU, Y. Understanding urban traffic flow characteristics: A rethinking of betweenness centrality. *Environment and Planning B: Planning and Design* (2012)
24. ZHAO, P.X., ZHAO S.M. Understanding urban traffic flow characteristics from the network centrality perspective at different granularities. *International Archives of the Photogrammetry Remote Sensing and Spatial Information Sciences* **41** (2016) 263 – 268
25. PIGGINS, A. Rationality for mortals: how people cope with uncertainty, by gerd gigerenzer. *The Journal of Positive Psychology* **7** (2012) 75 – 76
26. FREEMAN, L. C. A set of measures of centrality based on betweenness. *Sociometry* **40** (1977) 35 – 41
27. DIJKSTRA, E. W. A note on two problems in connexion with graphs. *Numerische Mathematik* **1** (1959) 269 – 271
28. BRANDES, U. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology* (2001)